

"Express Mail" Mailing Label No. EL437467382US

**PATENT APPLICATION
ATTORNEY DOCKET NO. SUN-P4671-AES**

5

**METHOD AND APPARATUS FOR CREATING
10 AN INDEX FOR A STRUCTURED DOCUMENT
BASED ON A STYLESHEET**

Inventor(s): Jacek R. Ambroziak

15

Related Applications

The application hereby claims priority under 35 U.S.C. § 119 to Provisional Patent Application No. 60/174,967 filed on January 6, 2000. The subject matter of this application is related to the subject matter in a co-pending non-provisional application by the same inventor as the instant application and filed on the same day as the instant application entitled, "Method and Apparatus for Flexibly Assigning Tokenization Procedures," having serial number TO BE ASSIGNED, and filing date TO BE ASSIGNED (Attorney Docket No. SUN-P4672-AES).

25

BACKGROUND

The present invention relates to indexing structures to facilitate computerized searches through data. More specifically, the present invention relates to a method

and an apparatus for generating an index to facilitate searching through data within a document based upon a predefined index stylesheet associated with the document that contains instructions for creating an index for the document.

5 The explosive growth of the Internet has been strongly tied to the development of search engines that allow users to rapidly search through large volumes of textual data from thousands and even millions of different web sites. A user who is interested in a particular topic merely has to enter a number of keywords into a search engine in order to receive linkages to different web pages containing the key words.

10 Search engines typically create an "index" of documents (such as web pages) that are available on the world wide web. An index typically stores individual words (or other meaning carrying textual strings) in a more compact and easily searchable form known as "tokens."

15 The process of building an effective index can be greatly complicated by the fact that documents can exist a wide variety of different forms which need to be indexed differently. For example, an efficient index for a technical paper might contain the abstract and title of the technical paper, but not the body of the technical paper, whereas an efficient index for a television schedule might contain ratings for individual television programs.

20 The process of creating an index is also complicated by the fact that for common document formats, such the Hypertext Markup Language (HTML) or the Extensible Markup Language (XML), much of the important information for search purposes is stored within attribute fields, and is not within the normal text of the document.

25 Furthermore, the structure of a document may change over time, which can require the structure of the index to change. For example, suppose the structure of

005220" 644E1S60
a product catalog is updated to include consumer reviews for individual products.
This change may require the index to change to include these consumer reviews.

Existing systems create indexes for documents using ad hoc rules. For example, one ad hoc rule is to create an index for all textual information that is not
5 within attribute fields. Unfortunately, such ad hoc rules often include much unimportant information in the index, and often exclude important information.

A similar problem exists in converting the document into tokens (tokenizing the document) during the index creation process. During the index creation process, relevant portions of a document are converted into tokens associated with individual
10 meaning-carrying units of text, such as wordforms or numbers. In the English language, wordforms are typically delineated by white spaces and punctuation marks. Hence, the tokenizing process is relatively easy. In contrast, languages such as Japanese have no such delineation. Consequently, the tokenization process depends on contextual information and can be very complicated.

15 The tokenization process can also be domain dependent. For example, periods within an email address, such as "person.dept@companyx.com" are linking elements, whereas periods within other textual information typically delineate word and sentence boundaries.

Hence, the tokenization process varies between languages and between
20 domains.

SUMMARY

One embodiment of the present invention provides a system that generates an index to facilitate searching through text within a document based upon an index
25 stylesheet associated with the document. The system operates by receiving a document to be indexed and then parses the document to produce a parsed document.

The system also retrieves instructions for creating the index for the document from an index stylesheet associated with the document. The system creates the index for the document by transforming the parsed document in a manner that is specified by the instructions retrieved from the index stylesheet.

5 In one embodiment of the present invention, retrieving the index stylesheet involves retrieving the index stylesheet across a network from a remote address.

 In one embodiment of the present invention, the index stylesheet is appended to the document.

 In one embodiment of the present invention, the system additionally makes
10 the index available to a search engine so that the search engine is able to scan through the index.

 In one embodiment of the present invention, the index stylesheet specifies sections of the document to skip in creating the index for the document.

 In one embodiment of the present invention, the index stylesheet specifies
15 attributes of the document that are to be included in the index.

 In one embodiment of the present invention, the system receives additional documents to be indexed, and creates indexes for the additional documents using the index stylesheet.

 In one embodiment of the present invention, creating the index for the
20 document involves tokenizing the document by partitioning text within the document into individual meaning-carrying units of text.

 In one embodiment of the present invention, prior to receiving the document, the system downloads and parses an index configuration file which specifies the index stylesheet to be used in creating the index.

25 In one embodiment of the present invention, the system receives the document from a client at an indexing server that creates the index for the client.

BRIEF DESCRIPTION OF THE FIGURES

FIG. 1 illustrates a distributed computing system in accordance with an embodiment of the present invention.

5 FIG. 2 illustrates how an indexing server creates an index for different document types in accordance with an embodiment of the present invention.

FIG. 3 illustrates how indexing stylesheets and tokenizing procedures are used to create and index for a document in accordance with an embodiment of the present invention.

10 FIG. 4 illustrates an example of the tokenization process.

FIG. 5 is a flow chart illustrating the process of creating an index in accordance with an embodiment of the present invention.

FIG. 6 is a flow chart illustrating the process of dynamically creating an updated index for an old document in accordance with an embodiment of the present invention.

15

DETAILED DESCRIPTION

The following description is presented to enable any person skilled in the art to make and use the invention, and is provided in the context of a particular application and its requirements. Various modifications to the disclosed embodiments will be readily apparent to those skilled in the art, and the general principles defined herein may be applied to other embodiments and applications without departing from the spirit and scope of the present invention. Thus, the present invention is not intended to be limited to the embodiments shown, but is to

20

25 be accorded the widest scope consistent with the principles and features disclosed herein.

The data structures and code described in this detailed description are typically stored on a computer readable storage medium, which may be any device or medium that can store code and/or data for use by a computer system. This includes, but is not limited to, magnetic and optical storage devices such as disk drives, magnetic tape, CDs (compact discs) and DVDs (digital video discs), and computer instruction signals embodied in a transmission medium (with or without a carrier wave upon which the signals are modulated). For example, the transmission medium may include a communications network, such as the Internet.

10 **Distributed Computing System**

FIG. 1 illustrates a distributed computing system 100 in accordance with an embodiment of the present invention. Distributed computing system 100 includes clients 102 and 118, which are coupled to indexing server 112 and search engine 122 through network 110.

15 Network 110 can include any type of wire or wireless communication channel capable of coupling together computing nodes. This includes, but is not limited to, a local area network, a wide area network, or a combination of networks. In one embodiment of the present invention, network 110 includes the Internet.

20 Clients 102 and 118 can include any node on network 110 including computational capability and including a mechanism for communicating across network 110.

Client 102 contains a number of documents 104-106, which are to be integrated into index 116 within indexing server 112. Indexing server 112 can include any node on a computer network including a mechanism for servicing requests from a client for computational and/or data storage resources. More

specifically indexing server 112 includes resources for creating index 116 within database 114 for indexing documents 104-106. Database 114 can include any type of mechanism for storing data in non-volatile form. In one embodiment of the present invention, database 114 includes the ORACLE 8TM database distributed by the Oracle Corporation of Redwood Shores, California.

Client 118 includes a browser 120 that communicates with search engine 122 in order to scan through index 116. Browser 120 can include any type of web browser capable of viewing a web site, such the INTERNET EXPLORERTM browser distributed by the Microsoft Corporation of Redmond, Washington.

Search engine 122 can include any type of computing system or application that is able to search through data.

During operation, indexing server 112 retrieves documents 104-106 from client 102, and uses documents 104-106 to create index 116. Note that client 102 may send documents 104-106 to indexing server 112. Or, alternatively, indexing server 112 can gather documents 104-106 from client 102.

Indexing server 112 creates index 116 by tokenizing selected portions of documents 104-106 and by creating index 116 from the tokens. Note that client 102 may itself be a server that makes documents 104-106 available over network 110.

After index 116 is created, client 118 submits a query 124 to search engine 122 through browser 120. Query 124 may specify keywords of interest to a user of client 118. In response to query 124, search engine 122 searches through index 116 to find documents containing matching keywords within index 116. If such documents are located, search engine 122 returns the documents in a list of query hits 126 to browser 120.

Indexing Server

FIG. 2 illustrates how indexing server 112 creates an index for different document types in accordance with an embodiment of the present invention. In FIG. 2, indexing server 112 receives many different document types from different sources. Indexing server 112 integrates these different document types into index 116. Note that index 116 may include a single index containing many different document types. Alternatively, index 116 may contain a different index for each document type.

An unlimited number of document types can be indexed. For example, FIG. 2 illustrates a news document 202, a product catalog 204, a television programming schedule 206, a document containing user documentation 208 and a document containing financial information 210. Each of these different document types can have a different document structure, which can be defined in a language such as XML. Each of these different document structures can be associated with a different indexing scheme. In some documents, specific attributes should be included in the index. For example, for some types of user documentation it may be advantageous to index an attribute that indicates whether the user documentation is suitable for a novice user or an expert user.

Indexing Stylesheets and Tokenization Procedures

FIG. 3 illustrates how stylesheets for indexing and procedures for tokenizing are used to create and index for a document in accordance with an embodiment of the present invention. In FIG. 3, an index building mechanism 310 within indexing server 112 takes as input a document 302 and produces an index 312 for the document 302. Index 312 feeds into a larger index 116 for a collection of documents, which is contained within database 114 (from FIG. 1).

005220" 6449 0360

During the index building process, index builder 310 refers to index stylesheet 304 and tokenizing procedures 306-307. Index stylesheet 304 contains a set of instructions for creating index 312 for document 302. For example, index stylesheet 304 can specify which sections of document 302 should be skipped in
5 producing an index 312. Index stylesheet 304 can also specify attributes of document 302 to be included in index stylesheet 304. For example, an attribute can specify a minimum security-level for a person to have access to document 302. Another attribute can specify a content rating (G, PG-13, PG, R, X) for document 302.

Tokenizing procedures 306-307 specify how certain portions of document
10 302 are to be tokenized. For example, tokenizing procedure 306 may specify how a first portion of document 302 is to be tokenized, while tokenizing procedure 307 specifies how a second portion of document 302 is to be tokenized. Although most documents will likely use of a single tokenizing procedure, other documents may includes sections in different languages or from different domains, which require
15 different tokenizing procedures.

Note that index stylesheet 304 is analogous to format stylesheets that are specified by the XML standard. Format stylesheets are used to specify display attributes, such as font and color, in order to display a XML document. Similarly, index stylesheet 304 specifies how an index for document 302 is to be created.

20 Note that although FIG. 3 illustrates indexing instructions and tokenizing instructions in the form of stylesheets and procedures, other representations are possible. For example, the indexing and tokenizing instructions can be contained within plug-in modules that can be plugged in to index builder 310.

These indexing and tokenizing instructions can also be referenced through an
25 object defined within an object-oriented programming system. For example, an

index parameter object can include a method that retrieves the instructions for building an index for document 302.

The tokenizing procedures 306-307 can additionally assume the form of code modules containing tokenization instructions, or may be provided by a remote service
5 across a network.

Also note that index stylesheet 304 may contain references specifying where tokenizing procedures 306-307 can be retrieved from.

Note that index builder 310 includes standardized interfaces that are able to accept input from many different index stylesheets and tokenizing procedures. This
10 allows index builder 310 to create indexes for many different document types using many different tokenization rules.

Tokenization Process

FIG. 4 illustrates an example of the tokenization process. In this example, a
15 piece of text "MAY 7, 2000" is divided into three tokens 402-404. Token 402 contains the word "MAY." Token 403 contains the day number "7," and token 404 contains the year number "2000." Each of these tokens is associated with a unique token number, and the unique token numbers are used to create the index. Using token numbers leads to a more compact representation because token numbers take
20 up less space than large character strings. Furthermore, during the searching process it is easier to lookup a number than a character string.

Process of Creating an Index

FIG. 5 is a flow chart illustrating the process of creating an index in
25 accordance with an embodiment of the present invention. The system starts by downloading a configuration file (step 502). This process may involve downloading

00549-02500
00549-02500

a configuration file across a network. Next, the system parses the configuration file (step 504) and identifies an address of an index stylesheet 304 within the configuration file (step 506).

5 Next, the system downloads the index stylesheet 304 from the identified address (step 508). This may involve retrieving the stylesheet across a network from a location specified by a universal resource locator (URL). Alternatively, the stylesheet can be appended to the document, in which case, the index stylesheet can be easily retrieved.

10 Next, the system parses the index stylesheet 304 so that the instructions within the index stylesheet can be used during the index creation process (step 510).

The system similarly identifies an address of a tokenizing procedure 306 (step 512). (Note that the address of tokenizing procedure 306 may be contained within the configuration file, or alternatively, within index stylesheet 304). Next, the system downloads tokenizing procedure 306 from the identified address (step 514). In one
15 embodiment of the present invention, tokenizing procedure 306 is retrieved from the same location as the index stylesheet 304. In another embodiment, tokenizing procedure 306 is retrieved from another location.

20 Next, the system inputs a number of documents into index 116. This is accomplished by downloading a document into index builder 310 (step 518) and then parsing the document using the instruction specified in index stylesheet 304 (step 520). The system converts the parsed document into tokens using tokenizing procedure 306 (step 522) and then creates an index using the tokens (step 524). This process is repeated for each document that is inputted into index 116.

25 After index 116 is complete, the system makes index 116 available to search engine 122 (from FIG. 1), so that search engine 122 can scan through index 116 to process queries (step 526).

FIG. 6 is a flow chart illustrating the process of dynamically creating an updated index for a document in accordance with an embodiment of the present invention. For certain types of searches data has a limited life. For example, in searching for current weather data, old weather data is of no interest.

5 In these circumstances, one embodiment of the present invention operates as follows. The system receives a search request (step 602). In response the search request, the system checks document creation dates for any documents that are involved in the search (step 604). If the system determines that a document is old (by perhaps comparing the age of the document against an age limit), the system causes
10 a new version of the document to be generated (for example, by gathering new weather data) (step 606). Next, the system creates an index for the new version of the document (step 608). This process implicitly removes the index for the old version of the document (step 610). Finally, the system performs a search involving the newly updated index.

15 The foregoing descriptions of embodiments of the invention have been presented for purposes of illustration and description only. They are not intended to be exhaustive or to limit the present invention to the forms disclosed. Accordingly, many modifications and variations will be apparent to practitioners skilled in the art.

For example, although the present invention is described in the context of a
20 distributed computing system including clients and servers, the present invention is not necessarily limited to a distributed client-server computing system. In general, the present invention can apply to any system that generates an index for textual information or any system that tokenizes textual information.

25 Additionally, the above disclosure is not intended to limit the present invention. The scope of the present invention is defined by the appended claims.